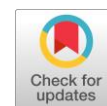


Detection and localization enhancement for satellite images with small forgeries using modified GAN-based CNN structure

Mohamed Mahmoud Fouad ^{a,1,*}, Eslam Magdy Mustafa ^{a,2}, Mohamed Abdelmoneim Elshafey ^{a,3}^a Department of Computer Engineering and Artificial Intelligence, Military Technical College, Cairo, Egypt¹ mmafoad@mtc.edu.eg; ² eng.eslam.mtc@gmail.com; ³ m.shafey@mtc.edu.eg

* corresponding author

ARTICLE INFO

Article history

Received September 8, 2020

Revised November 6, 2020

Accepted November 7, 2020

Available online November 30, 2020

Keywords

CNN

GAN

Satellite Image Forgery

Transfer Learning

FRReLU Activation Function

ABSTRACT

The image forgery process can be simply defined as inserting some objects of different sizes to vanish some structures or scenes. Satellite images can be forged in many ways, such as copy-paste, copy-move, and splicing processes. Recent approaches present a generative adversarial network (GAN) as an effective method for identifying the presence of spliced forgeries and identifying their locations with a higher detection accuracy of large- and medium-sized forgeries. However, such recent approaches clearly show limited detection accuracy of small-sized forgeries. Accordingly, the localization step of such small-sized forgeries is negatively impacted. In this paper, two different approaches for detecting and localizing small-sized forgeries in satellite images are proposed. The first approach is inspired by a recently presented GAN-based approach and is modified to an enhanced version. The experimental results manifest that the detection accuracy of the first proposed approach noticeably increased to 86% compared to its inspiring one with 79% for the small-sized forgeries. Whereas, the second proposed approach uses a different design of a CNN-based discriminator to significantly enhance the detection accuracy to 94%, using the same dataset obtained from NASA and the US Geological Survey (USGS) for validation and testing. Furthermore, the results show a comparable detection accuracy in large- and medium-sized forgeries using the two proposed approaches compared to the competing ones. This study can be applied in the forensic field, with clear discrimination between the forged and pristine images.

This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.

1. Introduction

The forensic community has developed many techniques to detect and localize image forgeries to address the problem of manipulated images, by means of some friendly programs, such as Photoshop [1]–[3]. Such developed techniques by forensics are one-sided towards imagery captured from purchaser cameras and cell phones that are completely different from sensors in satellites from the perspective of job description [4]–[9].

Generative adversarial networks (GANs) are presented as an effective technique to detect and localize forgeries in satellite images [10][11] that are based on convolutional neural networks (CNNs). For instance, in Wei *et al.* [12], a GAN is presented to relegate unlabeled information with virtual marks for advancing a training process; thus, a worldly based self-marking methodology for sheltered and stable information labeling is followed. The tested data set in Yarlagadda *et al.* [10] and Bartusiak *et al.* [11] includes satellite images [13][14] of different forgery sizes: small, medium, and large. In Yarlagadda *et*

al. [10] and *artusiak et al.* [11], the presented GAN-based approaches have a noticeable detection accuracy of 97% for large forgery size, however, with the lower detection accuracy of 79% with small forgeries. The satellite images feature high resolution. Hence, the real scenario of forgery is to be of small size that has been detected; however, with indeed low detection accuracy using GAN. As an approach for the detecting and localizing of satellite image forgery, the GAN depends mainly on the discriminator. This discriminator is based on a CNN model of a given structure and features. Therefore, it can be shown that the small forgeries are a significant challenge for their CNN-based discriminator [10][11].

In this paper, two different approaches are presented to improve the efficiency of the CNN-based discriminator as well as the whole GAN-based approach for detection and location. The first approach modifies the transfer learning technique in the trained discriminator of the GAN-based approach in *Yarlagadda et al.* [10]. That approach enables the detection accuracy detection to reach 86% for small forgeries. The second proposed approach includes a newly CNN-based discriminator. The structure and associated features of the proposed CNN, implemented in the discriminator of the second approach, increase the detection accuracy of small forgeries to 94%. The second proposed approach accomplishes 14% improvement in real scenarios of satellite images with small forgeries instead of the competing ones in *Yarlagadda et al.* [10] and *Bartusiak et al.* [11]. Comparative analysis between the two proposed approaches and the previously GAN-based presented in *Yarlagadda et al.* [10] and *Bartusiak et al.* [11] are performed in experiments with a unified dataset.

The paper is organized as follows. In Section 2, the related work to the satellite image forgery is demonstrated. In Section 3, the two proposed approaches are explained in detail. Experimental work and comparative analysis, including the dataset description as well as the evaluation metrics applied, are illustrated in Section 3. Finally, conclusions are drawn in Section 4.

2. Related Work

In this section, the state-of-the-art GAN-based models are thoroughly presented with detailed aspects of the discriminator mechanism.

In *Kohli et al.* [15], a hybrid technique is presented to detect and localize forged objects in images using temporal and spatial CNNs. A watermark-based method is shown to verify whether a satellite image is an authenticated one or not [16]. That method mainly depends on the embedded watermark, the absence of which makes that method ineffective. While, in *Gallego et al.* [17], the proposed method is based on one machine learning technique to detect forgeries in satellite images based on k-nearest neighbor (KNN) and artificial neural networks (ANN) algorithms. *Yarlagadda et al.* [10] and *Bartusiak et al.* [11] implemented two versions of GANs are presented in order to detect and localize forgeries of various sizes: small, medium, and large. The approaches provide higher accuracy of detection (i.e., ~97%) for satellite images with large-sized forgeries. However, those approaches provide a noticeable lower detection accuracy (i.e., ~79%) in small-sized forgeries, which is the real scenario.

In *Yarlagadda et al.* [10], a typical GAN is implemented to discriminate the pristine image from the forged and detect the forged pixels in case of forged image. As shown in *Fig. 1*, the architecture mainly includes two CNN-based models [10]: the generator and the discriminator. The generator is a CNN-based under complete Autoencoder that consists of two main parts: the encoder A_e and decoder A_d . The Autoencoder is trained by iterative procedures using gradient-descent-based minimization method with a cost function, such as

$$L = \mathcal{L}(x, A_d(A_e(x))) \quad (1)$$

The encoder A_e is a convolutional neural network that is symmetric to the deconvolutional neural network. The latter is considered as the decoder A_d , as shown in *Fig. 2*. Both are identical in the number of layers, each of five layers (i.e., in the encoder A_e : five convolutional layers from Conv1 through Conv5, while in the decoder A_d , other five deconvolutional layers from Deconv1 through Deconv5). In *Fig. 2*,

each Conv and Deconv layer is characterized by $z^*(x,y)$, where z is the count of the filters with the targeted layers, while x and y denote the 2D filter size.

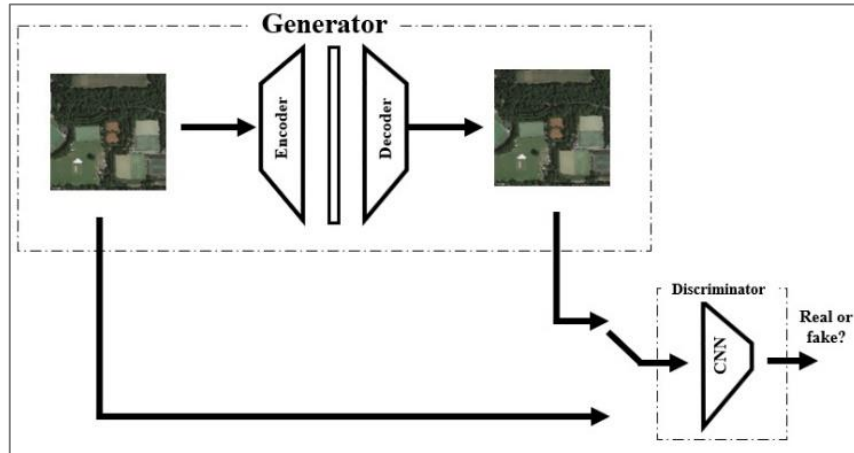


Fig. 1. General architecture of the implemented GAN [10][11].

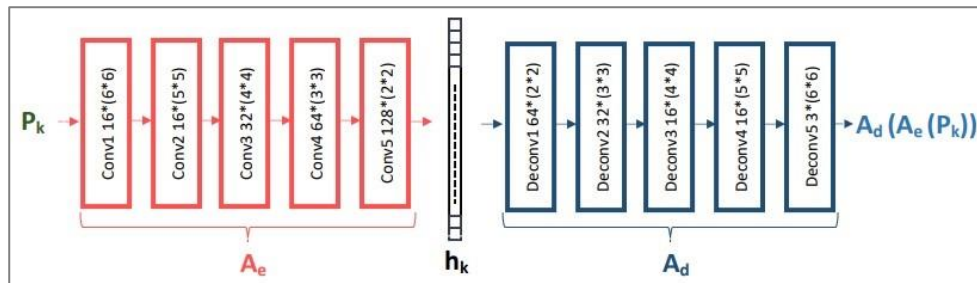


Fig. 2. The general architecture of the CNN-based autoencoder in the implemented GAN [10][11].

The output of the encoder is a feature vector, h_k , and is less than that of the input image in terms of dimensionality. A one-class of support vector machine (SVM) is fed with the vector h_k and learns from the pristine images (Fig. 3), then outputs a matrix P^* of the same input image size. P^* is an estimation of the binary matrix P of the pristine image, *i.e.*, all entries of P are all of the value of 1. Otherwise, entries with 0 values are pixel positions belonging to forgery in the image. As shown in Fig. 1, the main function of the discriminator is to precisely differentiate between real satellite images and those created by the generator that yields more real data as input.

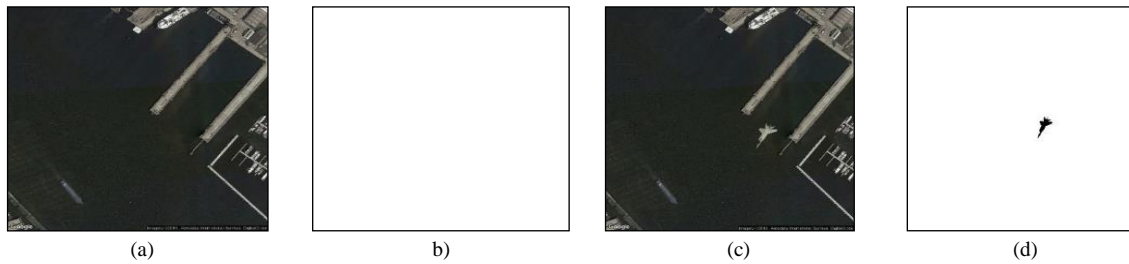


Fig. 3. A sample of a) pristine image, b) pristine mask, c) forged image, and d) forged mask

3. Method

In this section, the two proposed approaches are shown in Sections 3.1 and 3.2, respectively.

3.1. The first Proposed CNN-based Discriminator

The first proposed CNN-based discriminator has the same structure as that of the discriminator of the GAN-based model [10]. The structure of the discriminator is shown in Fig. 4. The GAN-based model in Yarlagadda *et al.* [10] provides a reasonable high detection accuracy in large and medium

forgeries and a significantly lower one with small forgeries, which is the real scenario. Each layer is followed by leaky rectified linear units (LReLU) activation function and a batch normalization (BN) layer, except the last convolutional layer that is followed by a single fully connected layer succeeded by a sigmoid layer for classification purpose [10].

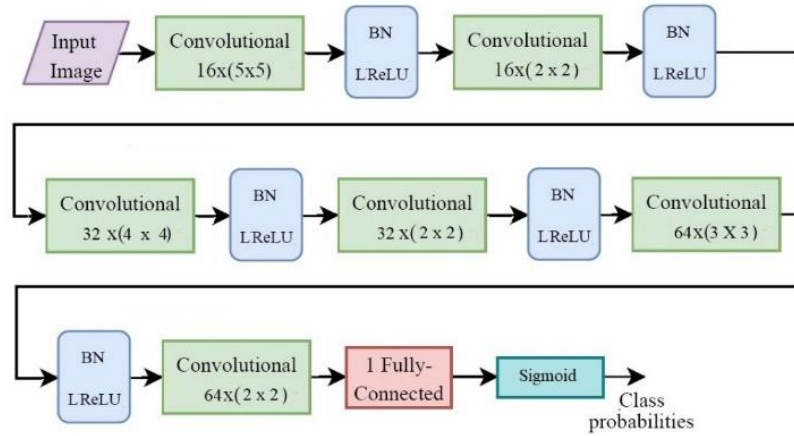


Fig. 4. The CNN structure of the implemented discriminator of the first proposed approach.

Generally speaking, the LReLU activation function is a mapping layer that transforms its input, i.e., the feature maps, through a nonlinear function to help the CNN improve the features' complexity before extraction and to map nonlinearly from input to output. The activation function has been differentiable to allow the CNN parameters to be optimized by the backpropagation strategy [18]. The mathematical form of the FReLU activation function, presented in Zhang *et al.* [19], is cast as in Eq. (2), and its curve is shown in Fig. 5. The LReLU activation function is not differentiable at the origin, thus leading to absurdly changing the values during the backpropagation step. Furthermore, the selection of a fixed value of the leak parameter prior to the training process would result in a non-optimal value [18].

$$f(x) = \begin{cases} \alpha x, & x < 0 \\ x, & x \geq 0 \end{cases} \quad (2)$$

The batch normalization (BN) layer performs normalization to the feature maps input in order to avoid the local minima problem. In addition, this layer improves the step of parameters updating and increase the invulnerability of initialized parameters. Finally, that layer also helps reach the convergence faster to force input data to be within the immersion areas [20].

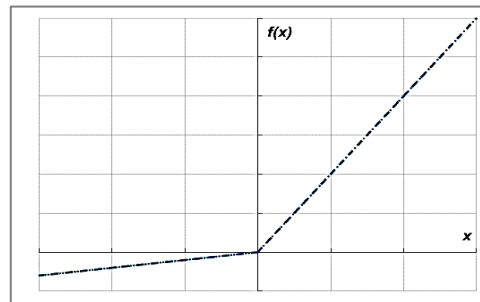


Fig. 5. Functional curve of the LReLU activation function [19].

The transfer learning is cast as an optimization problem, in which the learning enhancement of a new task can be achieved by knowledge transfer from a previously learned one [21]–[24], as shown in Fig. 6. In the first proposed approach, the pre-trained CNN-based discriminator's learned features in Yarlagadda *et al.* [10] and Bartusiak *et al.* [11] are transferred to a new CNN-based discriminator of the same structure. Then, that CNN is trained again using the dataset of satellite images of small-sized forgeries. This structure consists of 6 convolutional layers with different kernels number and kernel sizes, as shown in Table 1. Although the first proposed approach would enhance the detection accuracy as

opposed to those in Yarlagadda *et al.* [10] and Bartusiak *et al.* [11], we believe changing the CNN structure itself, in a certain manner, would lead to significant enhancement in the detection accuracy, as shown in the next section.

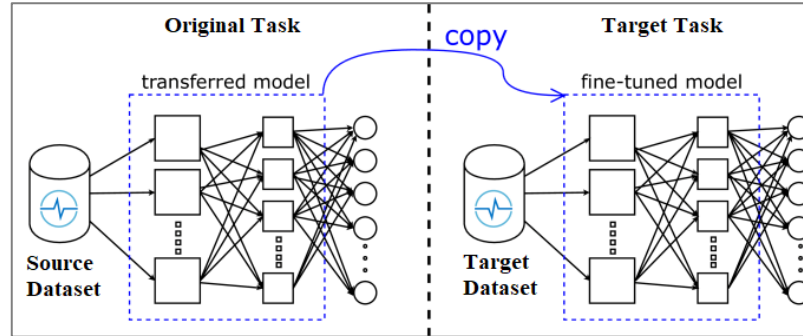


Fig. 6. Illustrating the transfer learning method [23].

Table 1. The Convolutional Layers of The CNN-Based Discriminator of the First Proposed Approach.

Layer	Count of Kernels	Kernel Size
1 st convolution Layer	16	5 X 5
2 nd convolution Layer	16	2 X 2
3 rd convolution Layer	32	4 X 4
4 th convolution Layer	32	2 X 2
5 th convolution Layer	64	3 X 3
6 th convolution Layer	64	2 X 2

3.2. The Second Proposed CNN-based Discriminator

In the second proposed approach, a completely different CNN-based discriminator in the GAN-based detection and localization is proposed, as in Fig. 7. Both 1st and second layers are followed by the TanH activation function [25] and batch normalization (BN) layers [20], which are succeeded by the average pooling layer (Avg Pooling) [26] in the case of the second convolutional layer. Each layer from the third through the fifth convolutional layers is followed by a single BN layer and single FReLU activation function [27], then succeeded by a maximum pooling layer (Max Pooling) [28]. The Max pooling layer is followed by a single fully connected layer, then by a sigmoid layer as in the first proposed approach.

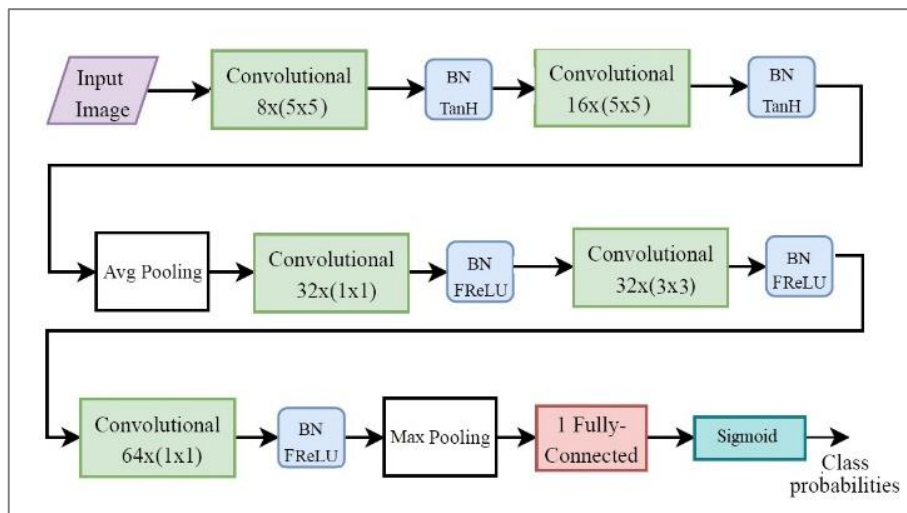


Fig. 7. The CNN structure of the implemented discriminator of the second proposed approach.

The second proposed CNN-based discriminator contains different key enhancements in order to result in better accuracy detection in case of small size forgeries. The second proposed approach is different from the first proposed one on three fronts as follows: 1) implementing both FReLU and Tanh activation functions, 2) adding different pooling layers, and 3) implementing a cyclic learning rate (CLR) method, instead of the fixed learning rate, during the training step. Such modifications are explained in detail as follows.

3.2.1. Flexible Rectified Linear Units (FReLU) activation function

The rectified point of Flexible Rectified Linear Units (FReLU) activation function can regulate the output using the negative data, and push the activation means closer to zero that speeds up the learning process [27]. That flexible rectification can enhance the CNN capacity. It features faster convergence and lower computations with higher performance. This function also is highly compatible with the BN layer [20]. The mathematical form of FReLU is cast as in Eq. 3, whereas its curve is shown in Fig. 8. The structure consists of five convolutional layers. The kernel size and its count are different for each convolution layer, as shown in Table 2.

$$f(x) = \begin{cases} b_n & , \quad x < 0 \\ x + b_n & , \quad x \geq 0 \end{cases} \quad (3)$$

Table 2. The Convolutional Layers of the CNN-Based Discriminator of the Second Proposed Approach.

Layer	Count of Kernels	Kernel Size
1 st convolution Layer	8	5 X 5
2 nd convolution Layer	16	5 X 5
3 rd convolution Layer	32	1 X 1
4 th convolution Layer	32	3 X 3
5 th convolution Layer	64	1 X 1

On the other hand, the Hyperbolic Tangent (Tanh) activation function (Fig. 9), generally named the Tanh function, converges faster than both sigmoid and logistic function, and provides better accuracy [29]. The Tanh seems to be slower than ReLU for many of the given examples but produces more natural fits for the data [30]. This is the main reason for implementing the Tanh function as an activation function after the first and second layers. Then, the FReLU activation function comes after the remaining layers.

$$f(x) = \frac{1-e^{-x}}{1+e^{-x}} \quad (4)$$

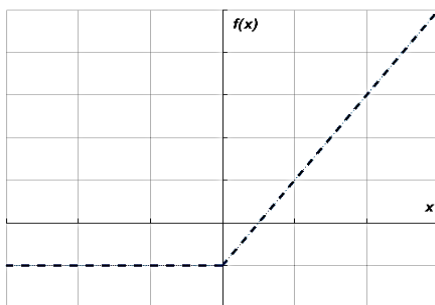


Fig. 8. Functional curve of FReLU activation function [27].

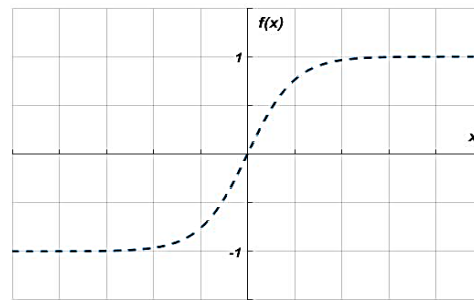


Fig. 9. Functional curve of Tanh activation function [29].

3.2.2. Pooling Layer

Pooling layer is simply used for reducing the feature map spatial dimension. Hence, this layer produces compact features that simplify the overall model complexity. This layer aims to obtain large

distance correlations of the input image, yielding vigorous matching of features even in case of small deformations [31]. The resultant compact representations of the feature maps help avoid overfitting. The second proposed approach implements two types of pooling layers: Maximum (Max) Pooling and Average (Avg) Pooling [31]. The Max pooling layer calculates the largest value in the batch of the feature map and then passes it to its output, as shown as the example in Fig. 10(a). While, with the average pooling layer, the average value of the batch is determined instead of choosing the largest number, as in Fig. 10(b). The Max pooling layer provides the most dominant features, such as edges, whereas the average pooling layer smoothly outputs the features [32]. As well, the Max pooling layer is better in coping with extracting the extreme features, whereas the average pooling layer brings all features into the count and delivers it to the next layer. This means that all values are actually used for feature mapping [28]. That is why the average pooling layer is implemented in the early layers, while the Max pooling layer is applied after the last convolutional layer.

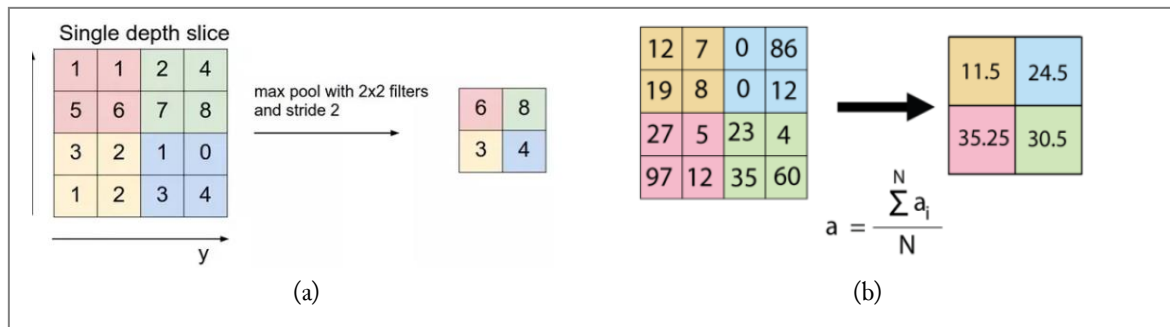


Fig. 10. Pooling case studies for (a) MaxPooling [28] , (b) AvgPooling [32].

3.2.3. The Cyclic Learning Rate (CLR)

The Cyclic Learning Rate (CLR) is one of the most considerable tuning hyper-parameters for training the deep NN-based models. The state-of-the-art stochastic gradient descent optimizer that is used for training purposes updates the weights based on loss function and the learning rate (LR). The training is robust for a low LR; however, the optimization consumes time because of the weight updates tiny values to reach the minimum loss function. On the other hand, for a higher LR, either the training would not converge, or the optimizer would be stuck with the local minima problem due to large values of the updates [33]. For each batch, the cyclic LR started from a low value and increased exponentially with recording the corresponding training loss [34]. Then, that loss is plotted against the corresponding LR within the interval [MinLR MaxLR], where MinLR is the minimum LR and MaxLR is the maximum LR. According to the faster decrease, the effective LR range will be selected among others, as shown in Fig. 11. Then, the training phase can be performed with a near-optimal learning rate.

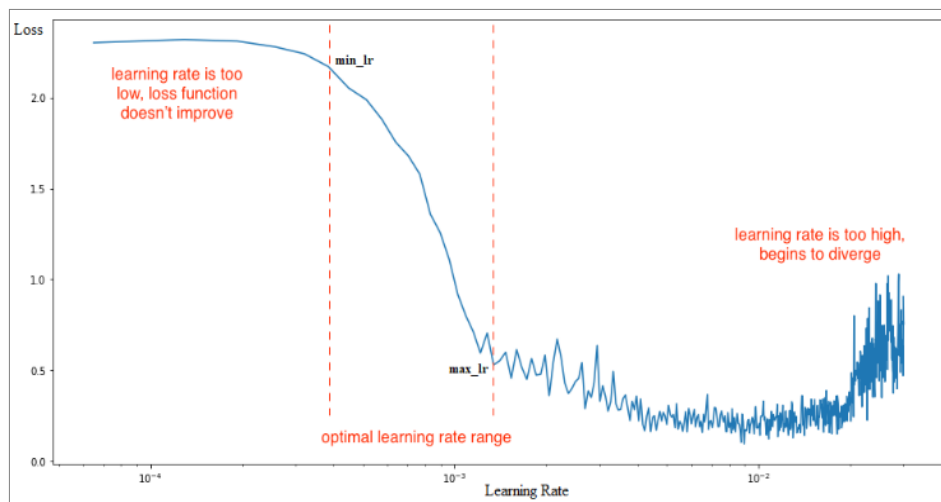


Fig. 11. Illustrating LR interval [Min_LR, Max_LR] [33][34].

4. Results and Discussion

In this section, the dataset upon, which experiments and comparative analysis are performed is described in Section 4.1. Then, the performance evaluation metrics and implementation setup are shown in Section 4.2. Finally, the experimental results and approaches' analysis are presented in Section 4.3.

4.1. Dataset Description

For a fair comparison, the dataset shown in Yarlagadda *et al.* [10] is typically used in all our experiments. The dataset used includes colored images with satellite scenes and the corresponding ground truth forgery binary images (i.e., masks). Those colored images are basically supported by the Landsat Science program [13], and then imposed in conjunction with those shown by NASA [14] and the US Geological Survey (USGS) [15]. The obtained dataset includes 344 image pairs of size 650x650 pixels. Objects, such as occlusions and airplanes (i.e., the masks), are basically spliced and imposed at random locations of the original colored images, yielding a dataset of forgery images.

That dataset includes 123 original image pairs (i.e., without any imposed forgery objects), and 221 forgery image pair. Those later are divided into three different sized-based forgery object categories, such as 158 pairs with small forgery objects, 32 with medium forgery objects, and 31 with large forgery objects. The size of the forgery objects is approximately 32x32, 64x64, and 128x128 pixels, respectively. For a fair comparison, some image geometric deformations, such as 90° and flipping, have been applied to the original and small-forgery image pairs to have the training data set size increased. Finally, the dataset includes four classes: S, M, L, and O classes, denoting the Small-, Medium-, Large-forgery image pairs and the Original image pairs, respectively, as shown in Fig. 12. Similarly, as shown in Yarlagadda *et al.* [10], the dataset is categorized into three separate partitions for training, validation, and testing. The training partition contains 128 S-pairs and 90 O-pairs. The validation partition includes 32 S-pairs and 18 O-pairs. The testing partition contains 32 M-, 31 L-, and 15 O-pairs. In this paper, we aim to enhance the overall detection accuracy and stress to improve the S-pairs, which is a clear challenge in the competing approach in Yarlagadda *et al.* [10].

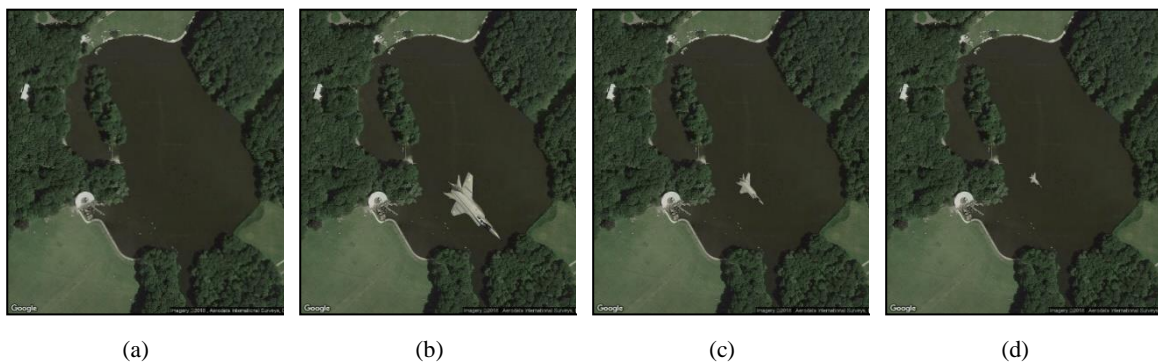


Fig. 12. Sample image from the implemented datasets [10][11] for a) Pristine sample of O dataset class, b) Forged sample of L dataset class, c) Forged sample of M dataset class, and d) Forged sample of S dataset class.

4.2. Performance Evaluation Metrics and Implementation Setup

In our experiments, we use three commonly used performance evaluation metrics: a) the detection accuracy; on the basis of the higher, the better, b) the receiver operator characteristic (ROC) curve, and c) the area under the curve (AUC); on the basis of the higher, the better. It worth noting that the ROC curve is a well-known evaluation metric for binary classification problems [35]. The ROC shows the relation between the true positive rate (TPR) and the false positive rate (FPR) of the ROC curve. The TPR denotes the correctly classified positive class subset, while the FPR denotes the incorrectly classified negative class with different thresholds [36][37]. The ROC curve is used to graphically show the trade-off between sensitivity and specificity for every possible cut-off for a test or a combination of tests [38]. Finally, the AUC is a metric that determines how well a classifier can differentiate between classes [39], [40], which is well suitable for our case.

All experiments are performed on a workstation with a CPU Intel Core i7 9th generation and 32 GB of RAM. Besides, the workstation is supported with GPU GTX 1650, 896 CUDA cores, a base clock of 1485 MHz, and a boost of up to 1665 MHz. The two proposed approaches are compared to those in Yarlagadda *et al.* [10] and Bartusiak *et al.* [11].

4.3. Experimental Results

Experiments are performed, using the dataset, explained in Sec. 4.1, three times with the original GAN-based detection and localization model shown in Yarlagadda *et al.* [10] and Bartusiak *et al.* [11] as well as the two proposed approaches; denoted as “Proposed 1” and “Proposed 2”. The results are listed in Table 3, for the three different sized forgeries: Large, Medium, and Small.

Table 3. Detection accuracy of all competing approaches using the same dataset.

Approach	Large Forgeries	Medium Forgeries	Small Forgeries
Yarlagadda <i>et al.</i> [10]	0.972	0.920	0.797
Bartusiak <i>et al.</i> [11]	0.982	0.946	0.803
Proposed 1	0.974	0.941	0.862
Proposed 2	0.983	0.960	0.943

Fig. 13 shows the ROC curve and the AUC values associated with the two approaches: Proposed 1 and Proposed 2 for satellite images with small forgeries only, which is our challenge to tackle in this paper. While, in Fig. 14, it can be shown that enhancement ratios of the proposed approaches over the GAN-based detection and localization models in Yarlagadda *et al.* [10] and Bartusiak *et al.* [11] are as follows:

- In the case of small size forgeries, the second proposed approach enhances the detection accuracy with ratios of 14.6 and 14.0 compared to those in Yarlagadda *et al.* [10] and Bartusiak *et al.* [11], respectively.
- In the case of medium-size forgeries, the second proposed approach enhances the detection accuracy with ratios of 4.0 and 1.2 compared to those in Yarlagadda *et al.* [10] and Bartusiak *et al.* [11], respectively.
- In the case of large size forgeries, no detection accuracy enhancement can be noticed by the second proposed approach instead of those in Yarlagadda *et al.* [10] and Bartusiak *et al.* [11].

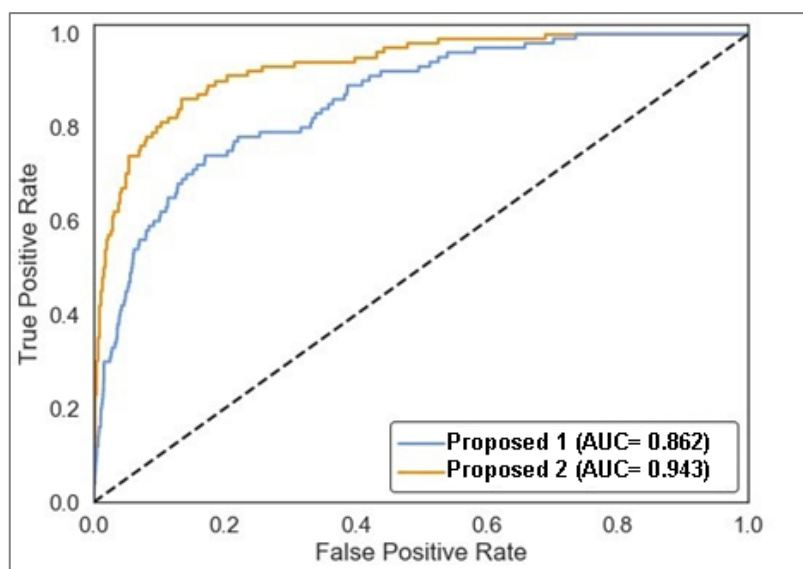


Fig. 13. The ROC curve of the approaches Proposed 1 and Proposed 2 for small forgeries.

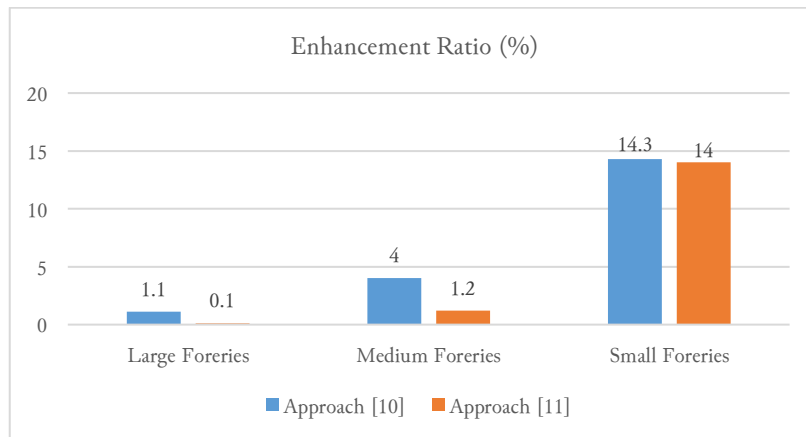


Fig. 14. Enhancement ratios of the second proposed approach over those approaches in [10] and [11].

5. Conclusion

This paper presents two GAN-based image forgery approaches. The first one uses the same CNN model, shown in Yarlagadda *et al.* [10]; however, with a completely different discriminator. The detection accuracy of the first proposed approach improvement reaches up to 6.5% as opposed to those in Yarlagadda *et al.* [10] and Bartusiak *et al.* [11], which can be further enhanced. In the second proposed approach, the CNN structure is entirely modified with a lower number of convolutional layers to have higher detection accuracy. Nevertheless, two other activation functions, namely the FReLU and Tanh, are used in conjunction with the cyclic learning rate for faster convergence and higher detection accuracy. In the case of small forgeries, which is the main of this paper challenge, as well as the approaches in Yarlagadda *et al.* [10] and Bartusiak *et al.* [11], shows that the second proposed approach outperforms the first proposed and those in Yarlagadda *et al.* [10] and Bartusiak *et al.* [11] by an average detection accuracy increase of 8.1%, 14.0%, and 14.6%, respectively. However, a slight enhancement in the detection accuracy is noticed using the second proposed approach compared to the competing approaches in large or medium forgeries.

Acknowledgment

The authors thank the anonymous reviewers for their valuable support and effort that have greatly improved the manuscript quality. As well, the authors thank the Department of Computer Engineering and Artificial Intelligence at the Military Technical College (Cairo, Egypt) for their appreciated encouragement.

Declarations

Author contribution. All authors have equally contributed to this article. As well, all authors have read and approved the final version of the article.

Funding statement. None of the authors has received any funding from any institution or source.

Conflict of interest. The authors declare no conflict of interest.

Additional information. No additional information is available for this paper.

References

- [1] A. Roy, R. Dixit, R. Naskar, and R. S. Chakraborty, *Digital Image Forensics: Theory and Implementation*, 2020, vol. 755, doi: [10.1007/978-981-10-7644-2](https://doi.org/10.1007/978-981-10-7644-2).
- [2] L. Verdoliva, "Media Forensics and DeepFakes: An Overview," *IEEE J. Sel. Top. Signal Process.*, vol. 14, no. 5, pp. 910–932, Aug. 2020, doi: [10.1109/JSTSP.2020.3002101](https://doi.org/10.1109/JSTSP.2020.3002101).
- [3] L. Ghammam, K. Karabina, P. Lacharme, and K. Thiry-Atighehchi, "A Cryptanalysis of Two Cancelable Biometric Schemes Based on Index-of-Max Hashing," *IEEE Trans. Inf. Forensics Secur.*, vol. 15, pp. 2869–2880, 2020, doi: [10.1109/TIFS.2020.2977533](https://doi.org/10.1109/TIFS.2020.2977533).

- [4] K. H. Rhee, "Detection of Spliced Image Forensics Using Texture Analysis of Median Filter Residual," *IEEE Access*, vol. 8, pp. 103374–103384, 2020, doi: [10.1109/ACCESS.2020.2999308](https://doi.org/10.1109/ACCESS.2020.2999308).
- [5] L. Bondi, S. Lameri, D. Guera, P. Bestagini, E. J. Delp, and S. Tubaro, "Tampering Detection and Localization Through Clustering of Camera-Based CNN Features," in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2017, pp. 1855–1864, doi: [10.1109/CVPRW.2017.232](https://doi.org/10.1109/CVPRW.2017.232).
- [6] R. Agarwal, D. Khudaniya, A. Gupta, and K. Grover, "Image Forgery Detection and Deep Learning Techniques: A Review," in *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)*, 2020, pp. 1096–1100, doi: [10.1109/ICICCS48265.2020.9121083](https://doi.org/10.1109/ICICCS48265.2020.9121083).
- [7] K. Saeed, S. Datta, and N. Chaki, "A Granular Level Feature Extraction Approach to Construct HR Image for Forensic Biometrics Using Small Training DataSet," *IEEE Access*, vol. 8, pp. 123556–123570, 2020, doi: [10.1109/ACCESS.2020.3006100](https://doi.org/10.1109/ACCESS.2020.3006100).
- [8] C. Chen, X. Zhao, and M. C. Stamm, "Generative Adversarial Attacks Against Deep-Learning-Based Camera Model Identification," *IEEE Trans. Inf. Forensics Secur.*, pp. 1–1, 2019, doi: [10.1109/TIFS.2019.2945198](https://doi.org/10.1109/TIFS.2019.2945198).
- [9] O. Kliuiev, M. Mozhaiev, O. Uhrovetskyi, O. Mozhaiev, and E. Simakova-Yefremian, "Method of Forensic Research on Image for Finding Touch up on the Basis of Noise Entropy," in *2019 3rd International Conference on Advanced Information and Communications Technologies (AICT)*, 2019, pp. 76–79, doi: [10.1109/AIACT.2019.8847760](https://doi.org/10.1109/AIACT.2019.8847760).
- [10] S. K. Yarlagaadda, D. Güera, P. Bestagini, F. Maggie Zhu, S. Tubaro, and E. J. Delp, "Satellite Image Forgery Detection and Localization Using GAN and One-Class Classifier," *Electron. Imaging*, vol. 2018, no. 7, pp. 214-1-214-9, Jan. 2018, doi: [10.2352/ISSN.2470-1173.2018.07.MWSF-214](https://doi.org/10.2352/ISSN.2470-1173.2018.07.MWSF-214).
- [11] E. R. Bartusiak *et al.*, "Splicing Detection and Localization In Satellite Imagery Using Conditional GANs," in *2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, 2019, pp. 91–96, doi: [10.1109/MIPR.2019.00024](https://doi.org/10.1109/MIPR.2019.00024).
- [12] X. Wei, X. Wei, W. Xing, S. Lu, and W. Lu, "An Incremental Self-Labeling Strategy for Semi-Supervised Deep Learning Based on Generative Adversarial Networks," *IEEE Access*, vol. 8, pp. 8913–8921, 2020, doi: [10.1109/ACCESS.2020.2964315](https://doi.org/10.1109/ACCESS.2020.2964315).
- [13] Landsat Science, "National Aeronautics and Space Administration," 2020. [Online]. Available: <https://landsat.gsfc.nasa.gov/>. [Accessed: 01-Jul-2020].
- [14] Science for a changing world, "U.S. Geological Survey, USGS.gov," 2020. [Online]. Available: <https://www.usgs.gov/>. [Accessed: 01-Jul-2020].
- [15] A. Kohli, A. Gupta, and D. Singhal, "CNN based localisation of forged region in object-based forgery for HD videos," *IET Image Process.*, vol. 14, no. 5, pp. 947–958, Apr. 2020, doi: [10.1049/iet-ipr.2019.0397](https://doi.org/10.1049/iet-ipr.2019.0397).
- [16] Xiaofeng Wang, Kemu Pang, Xiaorui Zhou, Yang Zhou, Lu Li, and Jianru Xue, "A Visual Model-Based Perceptual Image Hash for Content Authentication," *IEEE Trans. Inf. Forensics Secur.*, vol. 10, no. 7, pp. 1336–1349, Jul. 2015, doi: [10.1109/TIFS.2015.2407698](https://doi.org/10.1109/TIFS.2015.2407698).
- [17] A.-J. Gallego, J. Calvo-Zaragoza, and J. R. Rico-Juan, "Insights Into Efficient k-Nearest Neighbor Classification With Convolutional Neural Codes," *IEEE Access*, vol. 8, pp. 99312–99326, 2020, doi: [10.1109/ACCESS.2020.2997387](https://doi.org/10.1109/ACCESS.2020.2997387).
- [18] J. Pomerat, A. Segev, and R. Datta, "On Neural Network Activation Functions and Optimizers in Relation to Polynomial Regression," in *2019 IEEE International Conference on Big Data (Big Data)*, 2019, pp. 6183–6185, doi: [10.1109/BigData47090.2019.9005674](https://doi.org/10.1109/BigData47090.2019.9005674).
- [19] X. Zhang, Y. Zou, and W. Shi, "Dilated convolution neural network with LeakyReLU for environmental sound classification," in *2017 22nd International Conference on Digital Signal Processing (DSP)*, 2017, pp. 1–5, doi: [10.1109/ICDSP.2017.8096153](https://doi.org/10.1109/ICDSP.2017.8096153).
- [20] M. M. Kalayeh and M. Shah, "Training Faster by Separating Modes of Variation in Batch-Normalized Models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 6, pp. 1483–1500, Jun. 2020, doi: [10.1109/TPAMI.2019.2895781](https://doi.org/10.1109/TPAMI.2019.2895781).
- [21] R. Ashraf *et al.*, "Region-of-Interest Based Transfer Learning Assisted Framework for Skin Cancer Detection," *IEEE Access*, vol. 8, pp. 147858–147871, 2020, doi: [10.1109/ACCESS.2020.3014701](https://doi.org/10.1109/ACCESS.2020.3014701).

- [22] H. Fan, S. Gao, X. Zhang, X. Cao, H. Ma, and Q. Liu, "Intelligent Recognition of Ferrographic Images Combining Optimal CNN With Transfer Learning Introducing Virtual Images," *IEEE Access*, vol. 8, pp. 137074–137093, 2020, doi: [10.1109/ACCESS.2020.3011728](https://doi.org/10.1109/ACCESS.2020.3011728).
- [23] K. Bu, Y. He, X. Jing, and J. Han, "Adversarial Transfer Learning for Deep Learning Based Automatic Modulation Classification," *IEEE Signal Process. Lett.*, vol. 27, pp. 880–884, 2020, doi: [10.1109/LSP.2020.2991875](https://doi.org/10.1109/LSP.2020.2991875).
- [24] B. Cui, X. Chen, and Y. Lu, "Semantic Segmentation of Remote Sensing Images Using Transfer Learning and Deep Convolutional Neural Network With Dense Connection," *IEEE Access*, vol. 8, pp. 116744–116755, 2020, doi: [10.1109/ACCESS.2020.3003914](https://doi.org/10.1109/ACCESS.2020.3003914).
- [25] S. Obla, X. Gong, A. Aloufi, P. Hu, and D. Takabi, "Effective Activation Functions for Homomorphic Evaluation of Deep Neural Networks," *IEEE Access*, vol. 8, pp. 153098–153112, 2020, doi: [10.1109/ACCESS.2020.3017436](https://doi.org/10.1109/ACCESS.2020.3017436).
- [26] W. Gong, H. Chen, Z. Zhang, M. Zhang, and H. Gao, "A Data-Driven-Based Fault Diagnosis Approach for Electrical Power DC-DC Inverter by Using Modified Convolutional Neural Network With Global Average Pooling and 2-D Feature Image," *IEEE Access*, vol. 8, pp. 73677–73697, 2020, doi: [10.1109/ACCESS.2020.2988323](https://doi.org/10.1109/ACCESS.2020.2988323).
- [27] S. Qiu, X. Xu, and B. Cai, "FReLU: Flexible Rectified Linear Units for Improving Convolutional Neural Networks," in *2018 24th International Conference on Pattern Recognition (ICPR)*, 2018, pp. 1223–1228, doi: [10.1109/ICPR.2018.8546022](https://doi.org/10.1109/ICPR.2018.8546022).
- [28] Z. Ma *et al.*, "Fine-Grained Vehicle Classification With Channel Max Pooling Modified CNNs," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 3224–3233, Apr. 2019, doi: [10.1109/TVT.2019.2899972](https://doi.org/10.1109/TVT.2019.2899972).
- [29] X. Shen, X. Zhang, L. Lan, Q. Liao, and Z. Luo, "Another Robust NMF: Rethinking the Hyperbolic Tangent Function and Locality Constraint," *IEEE Access*, vol. 7, pp. 31089–31102, 2019, doi: [10.1109/ACCESS.2019.2903309](https://doi.org/10.1109/ACCESS.2019.2903309).
- [30] J. Yang, D. Ruan, J. Huang, X. Kang, and Y.-Q. Shi, "An Embedding Cost Learning Framework Using GAN," *IEEE Trans. Inf. Forensics Secur.*, vol. 15, pp. 839–851, 2020, doi: [10.1109/TIFS.2019.2922229](https://doi.org/10.1109/TIFS.2019.2922229).
- [31] N. Akhtar and U. Ragavendran, "Interpretation of intelligence in CNN-pooling processes: a methodological survey," *Neural Comput. Appl.*, vol. 32, no. 3, pp. 879–898, Feb. 2020, doi: [10.1007/s00521-019-04296-5](https://doi.org/10.1007/s00521-019-04296-5).
- [32] C.-Y. Lee, P. Gallagher, and Z. Tu, "Generalizing Pooling Functions in CNNs: Mixed, Gated, and Tree," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 863–875, Apr. 2018, doi: [10.1109/TPAMI.2017.2703082](https://doi.org/10.1109/TPAMI.2017.2703082).
- [33] R. Aldana, L. Campos-Macias, J. Zamora, D. Gomez-Gutierrez, and A. Cruz, "Dynamic Learning Rate for Neural Networks: A Fixed-Time Stability Approach," in *2018 24th International Conference on Pattern Recognition (ICPR)*, 2018, pp. 1378–1383, doi: [10.1109/ICPR.2018.8546084](https://doi.org/10.1109/ICPR.2018.8546084).
- [34] E. Hoffer, I. Hubara, and D. Soudry, "Train longer, generalize better: closing the generalization gap in large batch training of neural networks," in *Advances in Neural Information Processing Systems*, 2017, pp. 1731–1741, available at: [Google Scholar](https://scholar.google.com/citations?view_op=view_citation&hl=en&user=8YUWwvYAAAAJ&citation_for_view=8YUWwvYAAAAJ:1731-1741).
- [35] H. Narasimhan and S. Agarwal, "Support Vector Algorithms for Optimizing the Partial Area under the ROC Curve," *Neural Comput.*, vol. 29, no. 7, pp. 1919–1963, Jul. 2017, doi: [10.1162/NECO_a_00972](https://doi.org/10.1162/NECO_a_00972).
- [36] Y. Jiang, "Receiver Operating Characteristic (ROC) Analysis of Image Search-and-Localize Tasks," *Acad. Radiol.*, vol. 27, no. 12, pp. 1742–1750, Dec. 2020, doi: [10.1016/j.acra.2019.12.020](https://doi.org/10.1016/j.acra.2019.12.020).
- [37] W. Ma and M. A. Lejeune, "A distributionally robust area under curve maximization model," *Oper. Res. Lett.*, vol. 48, no. 4, pp. 460–466, Jul. 2020, doi: [10.1016/j.orl.2020.05.012](https://doi.org/10.1016/j.orl.2020.05.012).
- [38] G. Zhang, Z. Wang, and H. Mei, "Sensitivity clustering and ROC curve based alarm threshold optimization," *Process Saf. Environ. Prot.*, vol. 141, pp. 83–94, Sep. 2020, doi: [10.1016/j.psep.2020.03.029](https://doi.org/10.1016/j.psep.2020.03.029).
- [39] A. Joki-Rokita and R. Topolnicki, "Estimation of the ROC curve from the Lehmann family," *Comput. Stat. Data Anal.*, vol. 142, p. 106820, Feb. 2020, doi: [10.1016/j.csda.2019.106820](https://doi.org/10.1016/j.csda.2019.106820).
- [40] L. Omar and I. Ivrisimtzis, "Using theoretical ROC curves for analysing machine learning binary classifiers," *Pattern Recognit. Lett.*, vol. 128, pp. 447–451, Dec. 2019, doi: [10.1016/j.patrec.2019.10.004](https://doi.org/10.1016/j.patrec.2019.10.004).